

WHAT IS CLAIMED IS:

- 1 1. A computer-implemented method for processing
2 encryption requests, said method comprising:
3 sending, in a computer system having a plurality of
4 processors that share a common memory wherein at least
5 two of the processors are dislike, an encryption
6 request from a first processor to a second processor;
7 receiving, at the second processor, the encryption
8 request;
9 reading data from the common memory into a local
10 memory corresponding to the second processor, wherein
11 the reading is performed by the second processor and
12 wherein the second processor's local memory is not
13 shared with the first processor;
14 executing, at the second processor, an encryption
15 process corresponding to the request, the encryption
16 process adapted to transform the data; and
17 writing the transformed data from the second processor
18 to the common memory.
- 1 2. The method as described in claim 1 further comprising:
2 reading, at the second processor, one or more special
3 nonvolatile registers, the special registers including
4 one or more encryption keys; and
5 using one or more of the encryption keys in the
6 encryption process.
- 1 3. The method as described in claim 1 wherein the sending
2 further comprises writing the request to a mailbox

3 that corresponds to the second processor and the
4 receiving further comprises checking the second
5 processor's mailbox from the second processor.

1 4. The method as described in claim 1 further comprising:
2 identifying an input data area in the common memory
3 from which the data is read and an output buffer area
4 to which the transformed data is written.

1 5. The method as described in claim 1 further comprising:
2 initializing the second processor prior to receiving
3 the request, the initializing further including:
4 reading, from the common memory, initialization
5 software code to be executed on the second
6 processor; and
7 authenticating the initialization software code.

1 6. The method as described in claim 5 wherein the
2 authenticating is performed by a routine stored in a
3 nonvolatile memory and wherein the executing of the
4 encryption process is only performed if the
5 initialization software code is successfully
6 authenticated.

1 7. The method as described in claim 6 further comprising:
2 reading, at the second processor, one or more special
3 nonvolatile registers, the special nonvolatile
4 registers including one or more encryption keys, after
5 the initialization software code is successfully
6 authenticated; and

7 restricting access to the special nonvolatile
8 registers from outside of the second processor.

1 8. The method as described in claim 1 wherein the reading
2 and writing steps are performed using DMA operations.

1 9. The method as described in claim 1 further comprising:
2 identifying the encryption process and an encryption
3 algorithm from a plurality of encryption processes and
4 encryption algorithms based upon the encryption
5 request; and
6 loading encryption software code corresponding to the
7 identified encryption process and the encryption
8 algorithm, the loading performed by reading the
9 encryption software code from the common memory to the
10 second processor's local memory.

1 10. The method as described in claim 1 wherein the
2 encryption process is selected from the group
3 consisting of a decryption function, an encryption
4 function, and an authentication function.

1 11. An information handling system comprising:
2 a plurality of heterogeneous processors;
3 a common memory shared by the plurality of
4 heterogeneous processors;
5 a first processor selected from the plurality of
6 processors that sends an encryption request to a
7 second processor, the second processor also being
8 selected from the plurality of processors;

9 a local memory corresponding to the second processor,
10 wherein the second processor's local memory is not
11 shared with other processors included in the plurality
12 of processors; and

13 an encryption process running in the second processor,
14 the encryption process effective to:

15 read data associated with the encryption request
16 from the common memory to the second processor's
17 local memory;

18 transform the data based on the encryption
19 request; and

20 write the transformed data from the second
21 processor's local memory to the common memory.

1 12. The information handling system as described in claim
2 11 further comprising software code effective to:

3 read, at the second processor, one or more special
4 nonvolatile registers, the special registers including
5 one or more encryption keys; and

6 use one or more of the encryption keys in the
7 encryption process.

1 13. The information handling system as described in claim
2 11 wherein the sending of the encryption request
3 further comprises software code effective to:

4 write the encryption request to a mailbox that
5 corresponds to the second processor; and

6 read, from the second processor, the encryption
7 request from second processor's mailbox.

1 14. The information handling system as described in claim
2 11 further comprising software code effective to:
3 identify an input data area in the common memory from
4 which the data is read and an output buffer area to
5 which the transformed data is written.

1 15. The information handling system as described in claim
2 11 further comprising software code effective to:
3 initialize the second processor prior to receiving the
4 request, the initializing further including:
5 read, from the common memory, initialization software
6 code to be executed on the second processor; and
7 authenticate the initialization software code.

1 16. The information handling system as described in claim
2 15 wherein the software code effective to authenticate
3 the initialization software code is performed by a
4 routine stored in a nonvolatile memory, wherein the
5 encryption process is only performed if the
6 initialization software code is successfully
7 authenticated.

1 17. The information handling system as described in claim
2 16 further comprising software code effective to:
3 read, at the second processor, one or more special
4 nonvolatile registers, the special nonvolatile
5 registers including one or more encryption keys, after
6 the initialization software code is successfully
7 authenticated; and

8 restrict access to the special nonvolatile registers
9 from outside of the second processor.

1 18. The information handling system as described in claim
2 11 further comprising:

3 a DMA controller associated with each of the plurality
4 of processors, wherein the second processor reads from
5 and writes to the common memory using DMA operations
6 performed by the second processor's DMA controller.

1 19. The information handling system as described in claim
2 11 further comprising software code effective to:

3 identify the encryption process and an encryption
4 algorithm from a plurality of encryption processes and
5 encryption algorithms based upon the encryption
6 request; and

7 load encryption software code corresponding to the
8 identified encryption process and the encryption
9 algorithm, the load performed by reading the
10 encryption software code from the common memory to the
11 second processor's local memory.

1 20. The information handling system as described in claim
2 11 wherein the encryption process is selected from the
3 group consisting of a decryption function, an
4 encryption function, and an authentication function.

1 21. A computer program product stored on a computer
2 operable media for processing encryption requests,
3 said computer program product comprising:

4 means for sending, in a computer system having a
5 plurality of processors that share a common memory
6 wherein at least two of the processors are dislike, an
7 encryption request from a first processor to a second
8 processor;

9 means for receiving, at the second processor, the
10 encryption request;

11 means for reading data from the common memory into a
12 local memory corresponding to the second processor,
13 wherein the means for reading is performed by the
14 second processor and wherein the second processor's
15 local memory is not shared with the first processor;

16 means for executing, at the second processor, an
17 encryption process corresponding to the request, the
18 encryption process adapted to transform the data; and

19 means for writing the transformed data from the second
20 processor to the common memory.

1 22. The computer program product as described in claim 21
2 further comprising:

3 means for reading, at the second processor, one or
4 more special nonvolatile registers, the special
5 registers including one or more encryption keys; and

6 means for using one or more of the encryption keys in
7 the encryption process.

1 23. The computer program product as described in claim 21
2 wherein the means for sending further comprises means
3 for writing the request to a mailbox that corresponds

4 to the second processor and the means for receiving
5 further comprises means for checking the second
6 processor's mailbox from the second processor.

1 24. The computer program product as described in claim 21
2 further comprising:

3 means for identifying an input data area in the common
4 memory from which the data is read and an output
5 buffer area to which the transformed data is written.

1 25. The computer program product as described in claim 21
2 further comprising:

3 means for initializing the second processor prior to
4 receiving the request, the initializing further
5 including:

6 means for reading, from the common memory,
7 initialization software code to be executed on
8 the second processor; and

9 means for authenticating the initialization
10 software code.

1 26. The computer program product as described in claim 25
2 wherein the means for authenticating is performed by a
3 routine stored in a nonvolatile memory and wherein the
4 means for executing of the encryption process is only
5 performed if the initialization software code is
6 successfully authenticated.

1 27. The computer program product as described in claim 26
2 further comprising:

3 means for reading, at the second processor, one or
4 more special nonvolatile registers, the special
5 nonvolatile registers including one or more encryption
6 keys, the means for reading performed after the
7 initialization software code is successfully
8 authenticated; and

9 means for restricting access to the special
10 nonvolatile registers from outside of the second
11 processor.

1 28. The computer program product as described in claim 21
2 wherein the means for reading and means for writing
3 steps performed using DMA operations.

1 29. The computer program product as described in claim 21
2 further comprising:
3 means for identifying the encryption process and an
4 encryption algorithm from a plurality of encryption
5 processes and encryption algorithms based upon the
6 encryption request; and
7 means for loading encryption software code
8 corresponding to the identified encryption process and
9 the encryption algorithm, the means for loading
10 performed by reading the encryption software code from
11 the common memory to the second processor's local
12 memory.

1 30. The computer program product as described in claim 21
2 wherein the encryption process is selected from the
3 group consisting of a decryption function, an
4 encryption function, and an authentication function.